

(19) KOREAN INTELLECTUAL PROPERTY OFFICE

KOREAN PATENT ABSTRACTS

(11) Publication number: 1020020006336 A
 (43) Date of publication of application: 19.01.2002

(21) Application number: 1020000039950
 (22) Date of filing: 12.07.2000

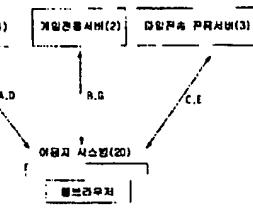
(71) Applicant: HAPPYTOGETHER CO., LTD.
 (72) Inventor: SEO, DONG HYEON

(51) Int. Cl. G06F 9/24

(54) METHOD FOR INSTALLING AUTOMATICALLY AND EXECUTING JAVA APPLET

(57) Abstract:

PURPOSE: A method for installing automatically and executing Java Applet is provided to improve the quality of network service and promote an advanced Internet communication by providing the high quality of service through Java technology.



CONSTITUTION: A network is comprised of a web server(1), a game server(2), a file server(3), and a user system(20). A web browser of the user system(20) connects to a web page of the web server(1), signs a user program for game to be downloaded, and releases a security(A). The web browser of the user system(20) downloads the user program through the file server(3)(B). After downloading, the web browser of the user system(20) executes the game by immediately connecting to the game server(2)(C). When the web browser of the user system(20) upgrades the program by connecting to the web server(1), the web browser downloads the web page only and the remaining executable file is directly executed on a local(D). When the web browser upgrades a file, the web browser downloads a corresponding file only(E). After downloading, the web browser of the user system(20) executes the game by connecting to the game server(2).

© KIPO 2002

Legal Status

Date of request for an examination (20000712)

Notification date of refusal decision (20030708)

Final disposal of an application (rejection)

Date of final disposal of an application (20030708)

(19) 대한민국특허청 (KR)
(12) 공개특허공보 (A)

(51) Int. Cl. 7
G06F 9/24

(11) 공개번호 특2002 - 0006336
(43) 공개일자 2002년01월19일

(21) 출원번호 10 - 2000 - 0039950
(22) 출원일자 2000년07월12일

(71) 출원인 해피투게더코퍼레이션 주식회사
서동현
서울 마포구 신수동 1 서강대창업보육센타 309

(72) 발명자 서동현
서울특별시마포구망원2동435 - 16지층2호

(74) 대리인 최홍순
박세걸
조성욱

심사청구 : 있음

(54) 자바 애플릿 자동 설치 및 실행 방법

요약

본 발명은 자바 애플릿으로 이미지, 오디오, 및 클래스 파일을 네트워크로 처음 로딩한 후에 그 다음부터는 업그레이드된 파일만 네트워크로 로딩하여 로컬에서 파일을 읽어 들여 수행하는 자바 애플릿 자동 설치 및 실행 기술에 관한 것이다. 본 발명은 이용자 시스템의 웹브라우저에서는 웹서버의 웹페이지에 접속하여 다운로드 받고자하는 자바 클래스 파일이 이용자 시스템의 클래스 패스에 존재하는지를 검토하는 단계; 검토하여 존재하지 않는 것으로 판명된 자바 클래스 파일을 이용자 시스템의 웹브라우저에서 읽어들이는 단계; 읽어들인 클래스 파일을 서명하여 자바 애플릿 보안을 해제하는 단계; 보안이 해제된 자바 클래스 파일을 파일 전송 서버를 통해 로컬 클래스 패스에 다운로드 받는 단계; 상기 단계에서의 다운로딩이 완료된 후에 자바 클래스 파일을 실행하는 단계; 웹서버와 접속하여 자바 클래스 파일을 업그레이드할 필요성이 생길 때, 웹페이지만 다운로드하고 나머지 실행파일은 로컬에서 바로 실행하는 단계; 업그레이드할 필요가 있는 자바 클래스 파일이 있을 경우에 해당 파일만 다운로드하는 단계; 및 다운로드 완료 후에 곧바로 이용자 파일 서버와 접속하여 자바 클래스 파일을 실행하는 단계를 포함한다.

대표도
도 3

색인어

자바 애플릿, 클래스 파일, 네트워크

명세서

도면의 간단한 설명

도 1은 종래의 보안 제한이 있는 애플릿을 사용하는 서비스 구성도.

도 2는 윈도우 응용프로그램을 사용하는 서비스 구성도.

도 3은 본 발명에 따른 서비스 구성도.

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

본 발명은 온라인 네트워크 서비스 기술에 관한 것으로, 특히, 자바 애플릿으로 이미지, 오디오, 및 클래스 파일을 네트워크로 처음 로딩한 후에 그 다음부터는 업그레이드된 파일만 네트워크로 로딩하여 로컬에서 파일을 읽어 들여 수행하는 자바 애플릿 자동 설치 및 실행 기술에 관한 것이다.

종래에 이 분야의 기술은 보안상 제한을 가지는 애플릿을 이용한 방법이나 윈도우 응용프로그램을 다운받아 실행하는 방법, 기타 플러그인을 설치하여 실행 파일을 다운받아 실행하는 방법 등이었다.

그러나, 종래의 자바 서비스는 애플릿을 이용해 이미지, 오디오 및 클래스 등의 파일을 네트워크로 매번 다운로드 받아 웹브라우저에서 실행을 해야 하며, 따라서, 약간 용량이 큰 자바 애플릿은 네트워크로부터 파일을 다운로드받아서 한번에 서비스를 제공받기 위해서는 상당한 시간을 필요로 한다. 이용자들은 매번 다운로드받을 때마다 상기와 같은 일을 반복해야 하므로 상당한 부담이 된다. 또한, 개발자 입장에선 이러한 문제 때문에 파일 사이즈에 제약을 받을 수밖에 없고, 이로 인해 서비스 질이 다른 윈도우 응용프로그램에 비해 상당히 떨어질 수밖에 없다.

또한, 윈도우 응용프로그램이나 플러그인 서비스의 경우는, 서비스를 받기 위해 일단 이용자가 응용프로그램 설치 파일을 다운로드하고, 다시 응용프로그램을 설치하므로 상당한 번거로움을 가지고 있다. 또한, 제대로 실행이 될 만한 프로그램은 그 파일의 용량이 10MB 내지 20MB를 초과하는 용량이기 때문에 쉽게 다운로드받아 실행할 엄두를 내지 못한다. 뿐만 아니라 이용자가 응용 프로그램을 실행하고자 하는 위치를 이동할 때마다 방대한 분량의 응용프로그램을 다시 다운로드받아야 하기 때문에 이용자들에게는 이 또한 상당한 부담으로 작용한다.

발명이 이루고자 하는 기술적 과제

따라서, 본 발명의 목적은 네트워크 상에서 이용자가 이용자 시스템에 가지고 있는 자바 애플릿 파일들의 정보를 서버의 최신정보와 비교해서 업그레이드될 필요성이 있는 자바 클래스 파일들만을 다운로드 시켜주는 기능과 동시에 개발자들의 개발 편의와 효율성을 위해 일반 FTP(File Transfer Protocol) 또는 윈도우탐색기에서 원하는 디렉토리에 파일을 이동시켜 자동으로 업그레이드 된 파일로그를 생성해주는 기능을 가지고 있는 새로운 구조의 파일서버를 구현하는데 있다.

본 발명의 또 하나의 목적은 웹브라우저마다 특성이 달라 제안을 받았던 각각의 자바 클래스 보안 인증 방법에 대한 종합분석을 통해 웹브라우저마다 특별한 설정 없이 동일하게 업그레이드 파일들만을 다운로드되도록 구현해 주는데 있다.

발명의 구성 및 작용

상기와 같은 목적을 달성하기 위하여, 본 발명은 이용자 시스템의 웹브라우저에서는 웹서버의 웹페이지에 접속하여 다운로드 받고자하는 자바 클래스 파일이 이용자 시스템의 클래스 패스에 존재하는지를 검토하는 단계; 검토하여 존재하지 않는 것으로 판명된 자바 클래스 파일을 이용자 시스템의 웹브라우저에서 읽어들이는 단계; 읽어들인 클래스 파일을 서명하여 자바 애플릿 보안을 해제하는 단계; 보안이 해제된 자바 클래스 파일을 파일 전송 서버를 통해 로컬 클래스 패스에 다운로드 받는 단계; 상기 단계에서의 다운로딩이 완료된 후에 자바 클래스 파일을 실행하는 단계; 웹서버와 접속하여 자바 클래스 파일을 업그레이드할 필요성이 생길 때, 웹페이지만 다운로드하고 나머지 실행파일은 로컬에서 바로 실행하는 단계; 업그레이드할 필요가 있는 자바 클래스 파일이 있을 경우에 해당 파일만 다운로드하는 단계; 및 다운로딩 완료 후에 곧바로 이용자 파일서버와 접속하여 자바 클래스 파일을 실행하는 단계를 포함하는 것을 특징으로 한다.

도 1은 네트워크상에서 종래의 보안제한이 있는 자바 애플릿을 사용하는 서비스 구성도이다. 도 1에 도시된 바와 같이, 웹서버(10)에서 이용자 시스템(20)에 존재하는 웹브라우저로 예를 들면, 자바로 작성된 게임 실행 파일을 다운로드 받아 게임을 운영하려 할 때, 이 게임 실행 파일인 자바 애플릿은 보안 제한이 있어 모든 프로그램 파일들이 하나의 물리적 웹서버(10) 내로 다운로드되고 또한 실행되어야 한다. 이로 인해 하나의 물리적인 웹서버(10)는 많은 부하가 가중되고 또한, 동일한 양의 모든 파일을 매번 다운로드받아야 하는 어려움이 있다.

이용자가 웹서버(10)에 접속할 때, 웹 문서에 자바 애플릿 클래스가 명시되어 있다면 이용자 시스템(20)의 웹브라우저에서는 명시된 자바 클래스 파일을 로딩하게 된다. 자바 클래스 파일을 로딩할 때 이용자 시스템(20)에 설정된 로컬의 클래스 패스에 먼저 접근한다. 그런 후 프로그램에 필요로 하는 자바 클래스를 로딩하여 실행을 하게 되는데 여기서 자바 클래스가 없으면 웹서버(10)에 있는 자바 클래스를 로딩하게 된다.

도 2는 네트워크 상에서 원도우 응용프로그램을 사용하는 서비스 구성도이다. 도 2에 도시된 바와 같이, 물리적으로 또는 기능적으로 분할된 웹서버(100), 게임 서버(200), 및 파일 전송 서버(300)와 인터넷 웹브라우저가 설치되어 있는 이용자 시스템(20)으로 구성된 네트워크 환경을 가정하자. 이용자 시스템(20)의 웹브라우저를 통하여 웹서버(100)에 처음 접속하여 예를 들면, 자바로 작성된 게임 전용 이용자 프로그램을 처리하는 과정은,

- A. 이용자 시스템(20)의 웹브라우저에서는 웹서버(100)의 웹페이지에 접속하여 게임 전용 이용자 프로그램의 다운로딩을 요청한다.
- B. 파일 전송 서버(300)를 통해 게임 전용 이용자 프로그램을 다운로드 받는다.
- C. B단계에서의 다운로딩이 완료된 후에 그 다운로드된 프로그램이 설치된다.
- D. 게임서버(200)와 접속하여 설치된 게임 프로그램이 실행된다.

이러한 과정 이후에 다시 웹서버와 접속하여 프로그램을 업그레이드할 필요성이 생길 때,

- E. 파일 전송 서버(300)와 접속하여 업데이트가 필요한 파일이 있을 경우 해당 파일만을 다운로드 받게 된다. 이 경우에 비교적 많은 양의 파일을 다운로드받게 된다.
- F. 게임 서버(200)와 접속하여 설치된 게임 프로그램이 실행된다.

도 2에서 설명된 바와 같이 이용자 프로그램을 처음에 웹서버(100)로부터 다운로드받은 후에 추가로 업그레이드할 파일을 다운로드 받는 것에 대한 원리를 설명하면 다음과 같다. 이용자가 웹서버(100)에 접속할 때, 웹 문서에 자바 애플릿 클래스가 명시되어 있다면 이용자 시스템(20)의 웹브라우저에서는 명시된 클래스 파일을 로딩하게 된다. 클래스 파일을 로딩할 때 이용자 시스템(20)에 설정된 로컬의 클래스패스에 먼저 접근한다. 그런 후 프로그램에 필요로 하는 클래스를 로딩하여 실행을 하게 되는데 여기서 클래스가 없으면 웹서버(100)에 있는 클래스를 로딩하게 된다.

이 기본 원리를 바탕으로 새로운 서비스 기술이 도출될 수 있다. 즉 처음에 네트워크로부터 한 개의 자바 클래스 파일을 다운로드받고 이용자는 이 파일에 대해서 보안 경고에 대한 인증을 받는다. 그리고, 실제로 서비스할 파일들을 네트워크로부터 전송을 받아 시스템 클래스 패스에 복사하여 놓으면 그 다음부터는 매번 네트워크로부터 다운로드받지 않고 로컬에서 빠르게 실행될 수 있는 것이다. 위와 같은 작업에 소요되는 시간은 기존의 평범한 일반 애플릿을 다운로드받아 실행하는 것과 똑같다. 왜냐하면 기존의 애플릿 또한 모든 파일들을 "Temp" 디렉토리에 복사해 넣고 실행을 한 뒤 실행이 끝나면 파일들을 삭제하는 방법을 쓰고 있기 때문이다. 그런데 기존의 애플릿은 이 작업을 매번 반복하는 것이다. 그러나 상기 기술을 쓴다면 같은 속도로 처음 한번만 수행하면 된다.

예를 들면, HelloApplet.class라는 자바 클래스 파일을 웹서버(100)에 놓으면 처음 방문하는 이용자 시스템(20)에서는 자바 클래스 패스를 검사하여 HelloApplet.class라는 자바 클래스 파일이 있는지를 검토하게 된다. 그 때 만약 HelloApplet.class라는 파일이 없다면 웹서버에 있는 이용자 시스템(20)의 웹브라우저는 클래스를 읽어 들인다. 이 때 처음 읽어 들이는 이 HelloApplet.class라는 파일을 캐비닛 파일에 넣고 서명을 하게 되면 자바 애플릿 보안을 해제할 수 있게 된다. 캐비닛 파일이 웹브라우저상에 로드되면 이 파일은 이용자에게 보안 경고라는ダイ얼로그 박스를 띄우게 되고, 이 보안 경고박스는 웹브라우저에서 실행될 클래스 파일이 로컬 파일에 접근하는 것을 금지하는 애플릿 보안에 대해 이용자에게 인증을 받고 실행하고자 하는 의도로써 이용자에게 인증을 요청한다는ダイ얼로그 박스로 이때 확인 버튼을 누르면 애플릿 보안을 해제해 준다. 이 보안경고ダイ얼로그 박스는 처음으로 접속하는 이용자에게 한번만 띄워주고 다음부터는 나타나지 않는다. 이용자에게 미리 이 보안 경고ダイ얼로그 박스에 대한 간단한 설명을 공고해 두어 처음 접속시 이 보안 경고ダイ얼로그 박스가 디스플레이되는 것이 이용자에게 크게 부담스럽지 않도록 할 수 있다.

그런 후에, 이용자가 확인 버튼을 누르면 보안이 해제된 HelloApplet 파일은 파일서버에 접속하여 실행에 필요한 클래스와 이미지, 오디오 등의 파일을 OS 설치 시 자동으로 설정된 로컬 클래스 패스에 다운로드를 받는다. 이 작업이 끝나게 되면 그 다음부터 웹브라우저는 상기 설명한 바와 같이, 웹페이지를 웹서버(100)에서 읽어 오지만 자바 클래스 파일은 로컬 클래스패스에 위치한 파일을 먼저 접근하게 되기 때문에 다운로드받은 자바 클래스파일을 읽어들인다. 이렇게 되면 다음부터는 캐비닛 파일을 읽는 것이 아니고 로컬에 있는 클래스파일을 읽는 것이므로 보안 경고ダイ얼로그 박스가 나타나지 않는다. 일단 이렇게 클래스파일이 설치되면 그 클래스들은 모두 보안이 해제된 상태에서 작동을 하게 된다. 그러므로 이후부터는 일반 웹페이지와 마찬가지로 부담 없이 하나의 웹페이지를 다운로드받게 되므로, 매번 모든 클래스, 오디오, 이미지 파일을 다운로드받을 필요 없이 이용자가 업그레이드될 필요성이 있는 한 두개 파일만 다운로드받아 대용량의 고품질 서비스를 아주 빠른 시간 안에 받을 수 있게 되는 것이다.

상기와 같은 원리를 기반으로 도 3에 도시된 바와 같은 하나의 실시예를 볼 수 있다. 도 3에 도시된 바와 같이, 물리적으로 또는 기능적으로 분할된 웹 전용서버(1), 게임 전용 서버(2), 및 파일 전송 전용 서버(3)와 인터넷 웹브라우저가 설치되어 있는 이용자 시스템(20)으로 구성된 네트워크 환경을 가정하자. 이용자 시스템(20)의 웹브라우저를 통하여 웹 전용 서버(1)에 처음 접속하여 자바로 작성된 게임 전용 이용자 프로그램을 처리하는 과정은,

A. 이용자 시스템(20)의 웹브라우저에서는 웹서버(1)의 웹페이지에 접속하여 다운로드받을 게임 전용 이용자 프로그램에 대해 서명하고, 보안 해제한다.

B. 파일 전송 전용 서버(3)를 통해 이용자 프로그램을 다운로딩 받는다.

C. B단계에서의 다운로딩이 완료된 후에 곧바로 게임 서버(2)와 접속하여 게임을 실행한다.

이러한 과정 이후에 다시 웹서버(1)와 접속하여 프로그램을 업그레이드할 필요성이 생길 때,

D. 웹페이지만 다운로드하고 나머지 실행파일은 로컬에서 바로 실행한다.

E. 업그레이드할 필요가 있는 파일이 있을 경우에 해당 파일만 다운로드한다. 이것은 아주 적은 사이즈의 파일 데이터이다.

F. 다운로드 완료후에 곧바로 게임 전용 서버(2)와 접속하여 게임을 실행한다.

본 발명에 따라서 상기와 같은 처리 순서로 매번 모든 클래스, 오디오, 이미지 파일을 다운로드받을 필요 없이 이용자가 업그레이드될 필요성이 있는 파일만 다운로드받아 기존의 서비스에 비해 대용량의 고품질 서비스를 아주 빠른 시간 안에 받을 수 있게 되는 것이다. 그 차이는 개발자에 따라 어느 정도 달라지지만 대략적으로 10배 이상의 차이가 난다. 예를 들어, 윈도우용용프로그램으로 20MB의 프로그램을 다운로드받는 경우라면 그와 같은 수준의 서비스를 제공하는 프로그램을 2MB만 다운로드받음으로써 해결할 수 있다는 것이다. 이것은 단순히 처음 다운로드받는 경우뿐만이 아니라 업그레이드를 받는 과정에서는 이보다 더욱 현격한 차이가 난다.

웹브라우저만 설치되어 있다면 불편한 인스톨 과정이 없다는 것도 좋은 이점이고 작업을 편리하게 할 수 있다.

또한, 자바 플랫폼의 독립적인 이점을 생각할 수 있다. 한번 개발된 프로그램으로 어느 플랫폼에서든 실행될 수 있고, 플랫폼이 바뀐다고 코드를 플랫폼에 맞추어 재작성할 필요가 없다는 것이다.

발명의 효과

본 발명에 따라서 이용자는 다른 윈도우용용프로그램과 마찬가지로 처음에 한번만 파일을 다운로드받으면 매번 파일을 다시 다운로드할 필요가 없어지며 윈도우 응용프로그램과 같이 방대한 양의 파일에 부담을 가질 필요도 없어지게 된다.

또한, 다른 프로그램 언어들보다 개발면에 있어 상당한 확장성과 효율성을 가지고 있는 자바 기술을 통해, 종래의 다른 서비스의 제약을 뛰어 넘을 수 있으며 손쉽게 고품질의 서비스를 제공할 수 있게 되므로 전반적인 네트워크 서비스의 질이 향상되어 진다. 이로 인한 진보된 인터넷 커뮤니케이션이 이루어지는데 많은 도움이 될 것으로 보여진다.

따라서, 본 발명은 종래의 온라인 네트워크 서비스의 단점 보완 및 개발효율성의 극대화와 이로 인한 서비스업체의 이윤창출을 제공할 수 있는 것이다.

(57) 청구의 범위

청구항 1.

이용자 시스템의 웹브라우저에서는 웹서버의 웹페이지에 접속하여 다운로드 받고자하는 자바 클래스 파일이 이용자 시스템의 클래스 패스에 존재하는지를 검토하는 단계; 검토하여 존재하지 않는 것으로 판명된 자바 클래스 파일을 이용자 시스템의 웹브라우저에서 읽어들이는 단계; 읽어들인 클래스 파일을 서명하여 자바 애플리 보안을 해제하는 단계; 보안이 해제된 자바 클래스 파일을 파일 전송 서버를 통해 로컬 클래스 패스에 다운로딩 받는 단계; 상기 단계에서의 다운로딩이 완료된 후에 자바 클래스 파일을 실행하는 단계; 웹서버와 접속하여 자바 클래스 파일을 업그레이드할 필요성이 생길 때, 웹페이지만 다운로드하고 나머지 실행파일은 로컬에서 바로 실행하는 단계; 업그레이드할 필요가 있는 자바

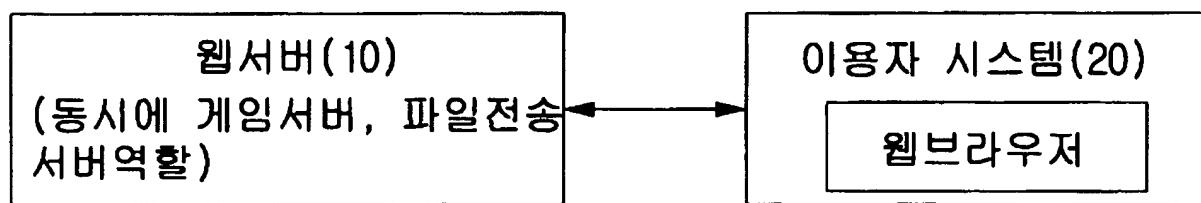
클래스 파일이 있을 경우에 해당 파일만 다운로드하는 단계; 및 다운로드 완료 후에 곧바로 이용자 파일 서버와 접속하여 자바 클래스 파일을 실행하는 단계를 포함하는 자바 애플릿 자동 설치 및 실행 방법.

청구항 2.

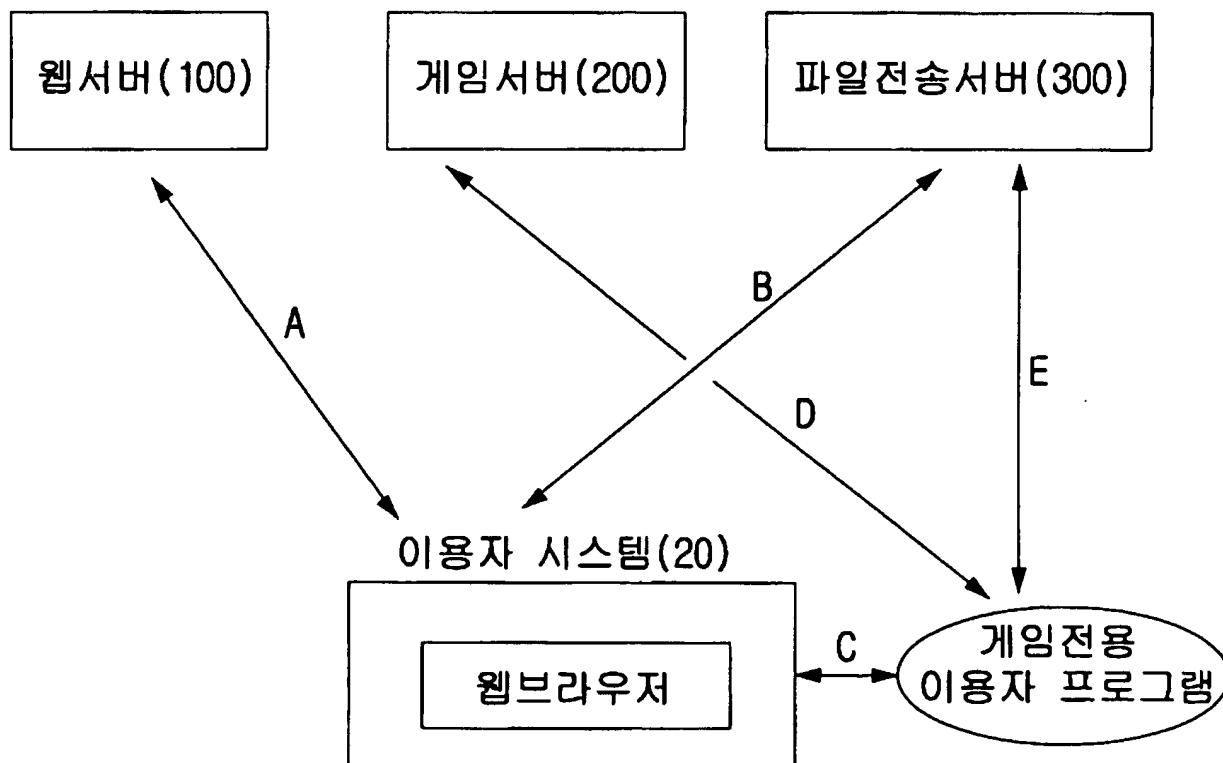
제 1 항에 있어서, 상기 이용자 시스템의 웹브라우저에서 읽어들인 클래스 파일을 서명하여 자바 애플릿 보안을 해제하는 단계에서, 웹브라우저에서 실행될 클래스 파일이 로컬 파일에 접근하는 것을 금지하는 애플릿 보안에 대해 이용자 인증으로 애플릿 보안을 해제할 수 있는 자바 애플릿 자동 설치 및 실행 방법.

도면

도면 1



도면 2



도면 3

